Delay and Complexity Tradeoffs for Dynamic Routing and Power Allocation in a Wireless Network

MIT -- Laboratory for Information and Decision Systems (LIDS)

Michael J. Neely
Jun Sun
Eytan Modiano

{mjneely@mit.edu, junsun@mit.edu, modiano@mit.edu}
Network Model:

Problems with the model:

- No Interference Effects: $\mu_{ab} = \mu_{ab}(p_{ab})$

- No Time Variation

- Fluid Model of data flow

What does the model capture?

- Nonlinear Power Allocation Problem

- Complexity of scheduling optimal strategy
2 ideas of this paper:

1. Capacity (100% thru-put) strategy obtained by iteratively solving a min-clearance time problem.

2. Complexity/Delay tradeoff by solving the min clearance problem over longer time intervals.
Min Clearance Problem:
No arrivals. Have backlog at time 0.

\( U_{ij} = \text{Unfinished bits in node } i \text{ (to be delivered to node } j) \).

Find routing and power controls \( p_{ij}(t) \) to clear in min time.

Observation: Optimal control can be restricted to constant power allocation strategies.
Proof sketch:

Given optimal \( p_{ij}(t) \) (clears in minimum time \( T \)).
Let \( \bar{p}_{ij} \) represent the empirical avg. during \([0, T]\).

\[
\frac{1}{T} \int_0^T \mu_{ij}(p_{ij}(\tau)) \, d\tau \leq \mu_{ij}(\bar{p}_{ij})
\]

(by concavity of \( \mu() \) and Jensen’s inequality)
From this, it is straightforward to form the min clearance time solution as a convex optimization problem:

Problem $\pi_{\text{min}}$

Maximize $\gamma$

Subject to: $f_{ij}^{(c)} \geq 0$

$$\sum_{a=1}^{N} f_{ai}^{(c)} - \sum_{b=1}^{N} f_{ib}^{(c)} = -\gamma U_{ic}^{c} + \delta_{i-c} \sum_{j=1}^{N} \gamma U_{jc}$$

$$\sum_{c=1}^{N} f_{ij}^{(c)} \leq \mu_{ij}(\bar{p}_{ij})$$

$$\sum_{j=1}^{N} \bar{p}_{ij} \leq P_{i}^{tot}$$
Dynamic Scheduling Using Iterative solution of $\pi_{min}$:

$U_{ij}[k] \quad U_{ij}[k+1]$

$T_k \quad T_{k+1}$

Iterative Minimum Emptying Time algorithm (IMET):

1. If the system is empty, wait for new data to enter.

2. Start iteration $k$ by observing the current backlog $U_{ij}[k]$, and solve $\pi_{min}$ for this backlog, clearing it in time $T_k$. Hold routing and scheduling fixed for duration $T_k$.

3. Repeat for iteration $k+1$. 
Let:
\[ \Lambda = \text{set of data rates } (\lambda_{ij}) \text{ the network can stably support.} \]

Can be shown that \( \Lambda \) is the set of all rates \( \lambda_{ij} \) such that there exists a constant power allocation \( p_{ij}^* \) for which a multi-commodity flow can be set up over the network (with link capacities \( \mu_{ij}(p_{ij}^*) \)) that satisfies the \( \lambda_{ij} \) rates.

Traffic Assumptions -- Time varying leaky bucket:

\[ X_{ij}(t) = \text{Bits arrived to node } i \text{ destined for } j \text{ during } [0, t]. \]

\[ X_{ij}(t + T) - X_{ij}(t) \leq \sigma + \int_t^{t+T} \lambda_{ij}(\tau) d\tau \]

where \( (\lambda_{ij}(t) + \epsilon) \in \Lambda \) for all \( t \)

\( \lambda_{ij}(t) = \text{instantaneous data rate of } X_{ij}(t) \text{ stream} \)

\( \sigma = \text{traffic burst parameter} \)

\( \epsilon = \text{distance the instantaneous data rate is from the boundary of the capacity region} \)

These above parameters are unknown to the network controller.
Theorem: The IMET Algorithm guarantees:

\[ T_{\text{worst-case}} \leq 2\sigma / \varepsilon \]

Proof:

\[
\begin{array}{c|c|c}
U_{ij}[k] & U_{ij}[k+1] \\
\hline
T_k & T_{k+1} \\
\end{array}
\]

Let \( \lambda_{ij} \) represent the rate of traffic during interval \( T_k \).
By assumption, there is a \( \lambda_{ij}^* \in \Lambda \) such that \( \lambda_{ij} + \varepsilon \leq \lambda_{ij}^* \).

\[
T_{k+1} = \min \text{ time to clear}
\]

\[
\leq \max_{(i,j)} \frac{U_{ij}}{\lambda_{ij}^*}
\]

\[
\leq \max_{(i,j)} \frac{\sigma + \lambda_{ij} T_k}{\lambda_{ij}^*}
\]

\[
\leq \max_{(i,j)} \frac{\sigma + (\lambda_{ij}^* - \varepsilon) T_k}{\lambda_{ij}^*}
\]

\[
\leq \frac{\sigma}{\varepsilon}
\]
Complexity Constraint:

The IMET algorithm requires the solution to a convex optimization to be computed instantaneously at the beginning of a slot.

Idea: Compute solution of $U_{ij}[k]$ problem during $T_{k+1}$.

Computational Processing Speed Constraint:
$C = \text{Processing Rate (floating point ops / second)}$

Let $a_N = \# \text{operations required to compute the solution of the convex optimization for a net. of size } N$.

Modified IMET:
- Shift computations by one interval $T_k$.
- Hold solutions fixed for $\max\{\text{emptying time, } a_N/C\}$
Theorem (for modified IMET):

\[
T_{\text{worst-case}} \leq 3 \max \left[ \frac{\sigma}{\varepsilon}, \frac{a_N}{C} \right]
\]

(compared to original IMET bound of \(2\sigma/\varepsilon\)).

Conclusions:
- Iterative Min Emptying Time algorithm IMET
- Acts without knowledge of rate or burst parameters \((\lambda_{ij}(t)), \varepsilon\)
- 100% throughput, Worst Case Delay Bound

Future Work... Time varying systems
  Fairness issues